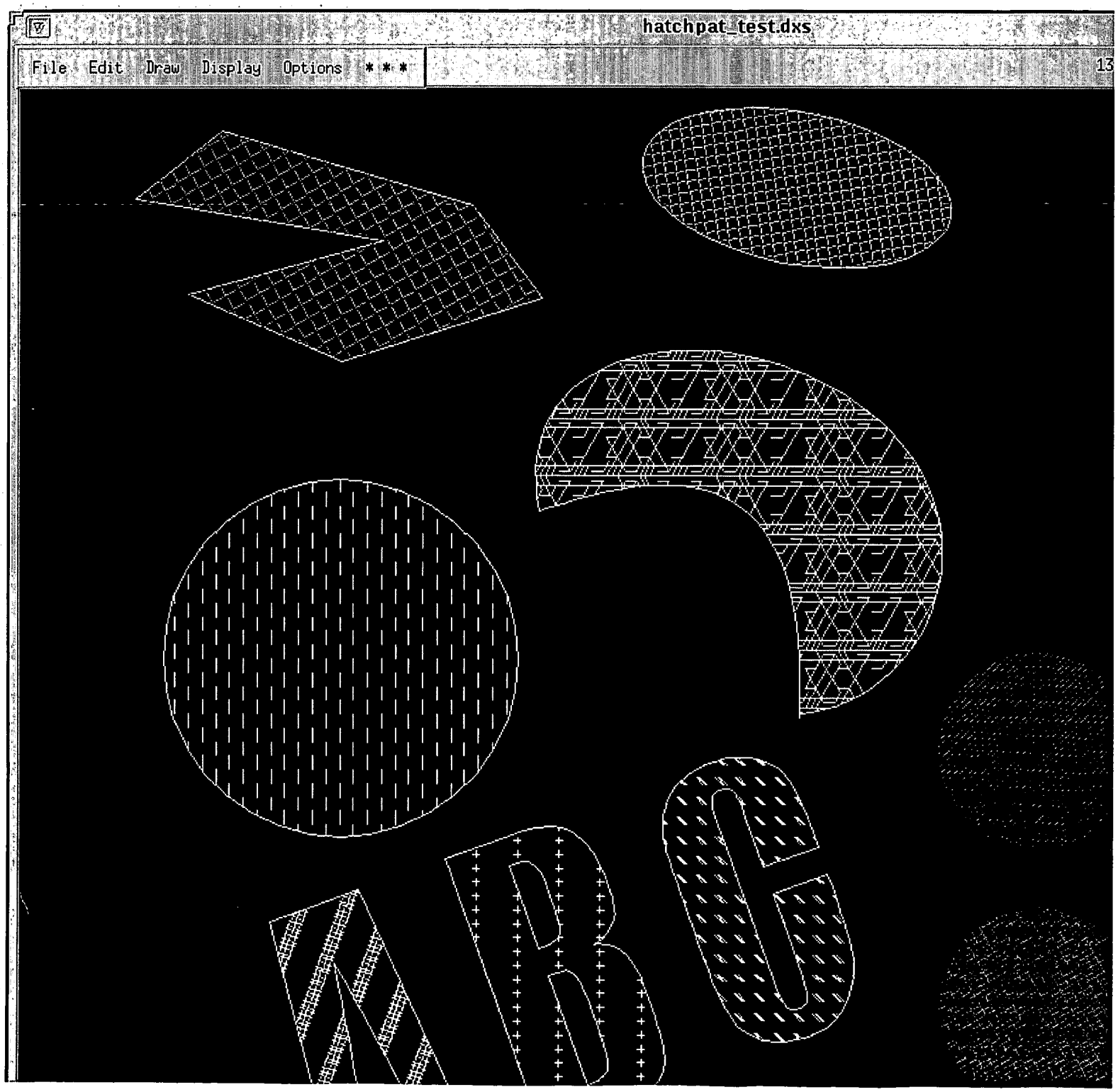
back to Home Page

Like leading drafting programs LinuxCAD implements various types of hatching, that includes hatching by simple lines and hatching by hatch patterns. You can reuse any and all AutoCAD hatch patterns in LinuxCAD.

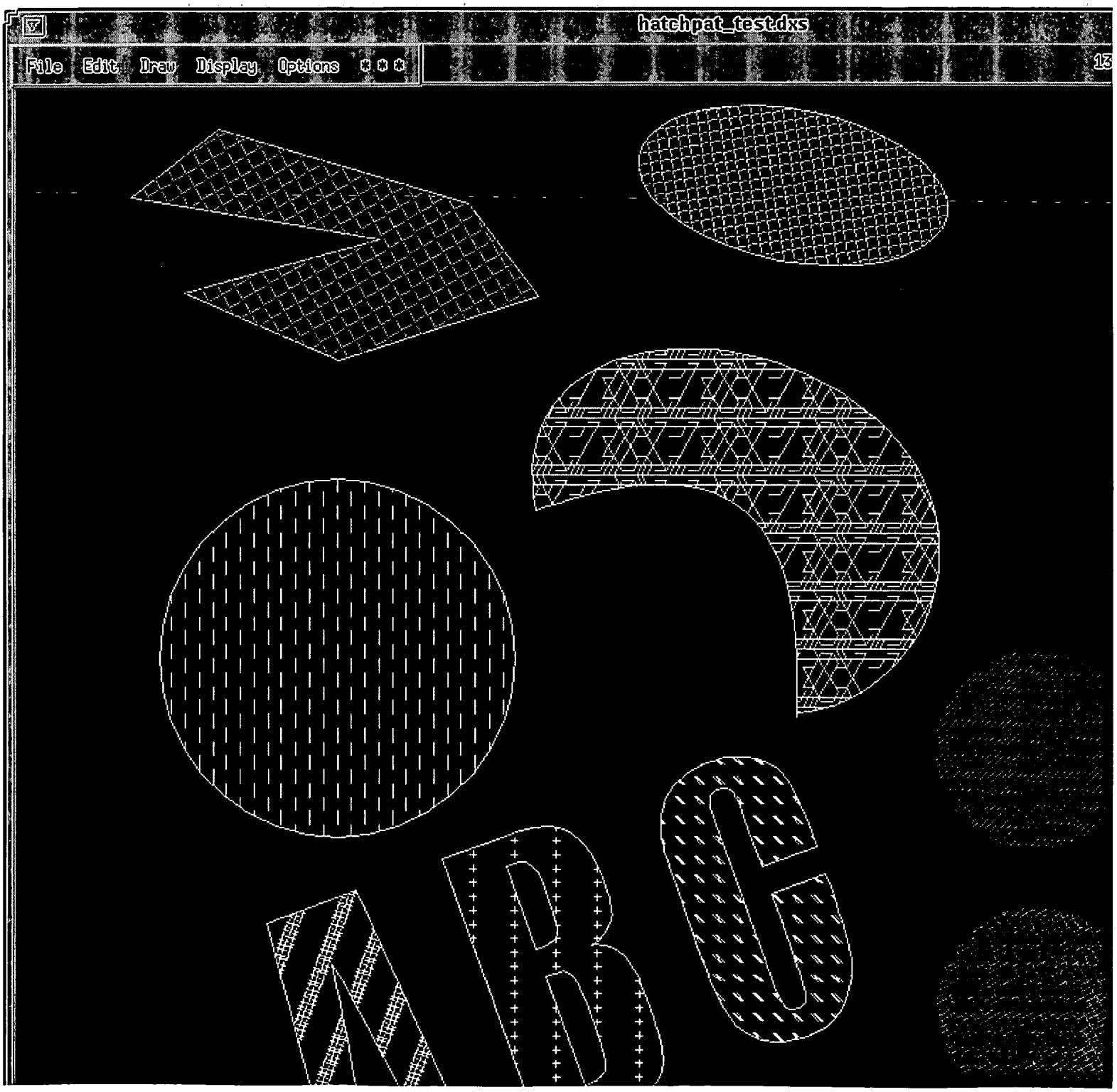
At least 80% of the drafting work people do in AutoCAD may be done in LinuxCAD instead !!!



back to Home Page

Like leading drafting programs LinuxCAD implements various types of hatching, that includes hatching by simple lines and hatching by hatch patterns. You can reuse any and all AutoCAD hatch patterns in LinuxCAD.

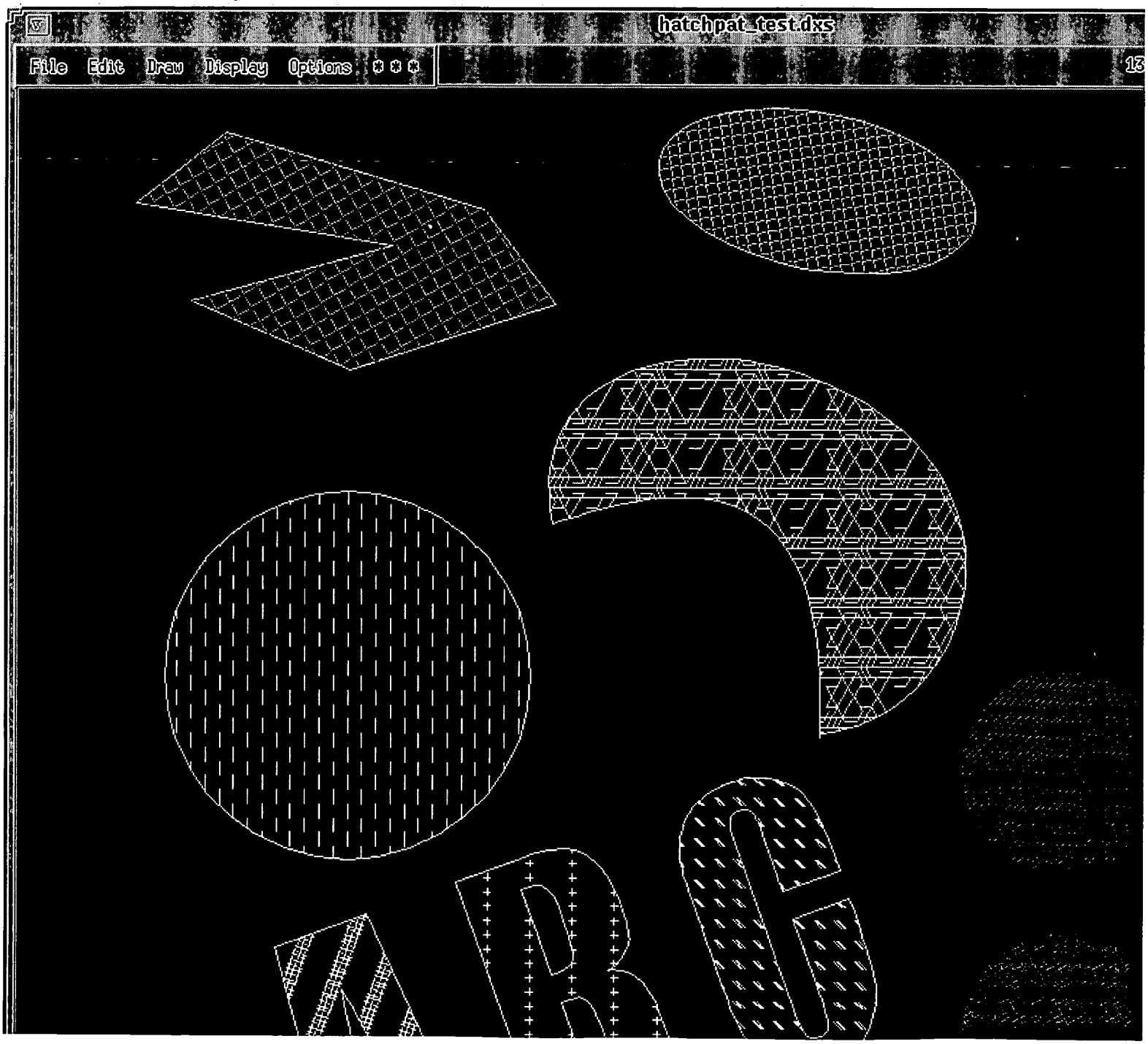
At least 80% of the drafting work people do in AutoCAD may be done in LinuxCAD instead !!!



back to Home Page

Like leading drafting programs LinuxCAD implements various types of hatching, that includes hatching by simple lines and hatching by hatch patterns. You can reuse any and all AutoCAD hatch patterns in LinuxCAD.

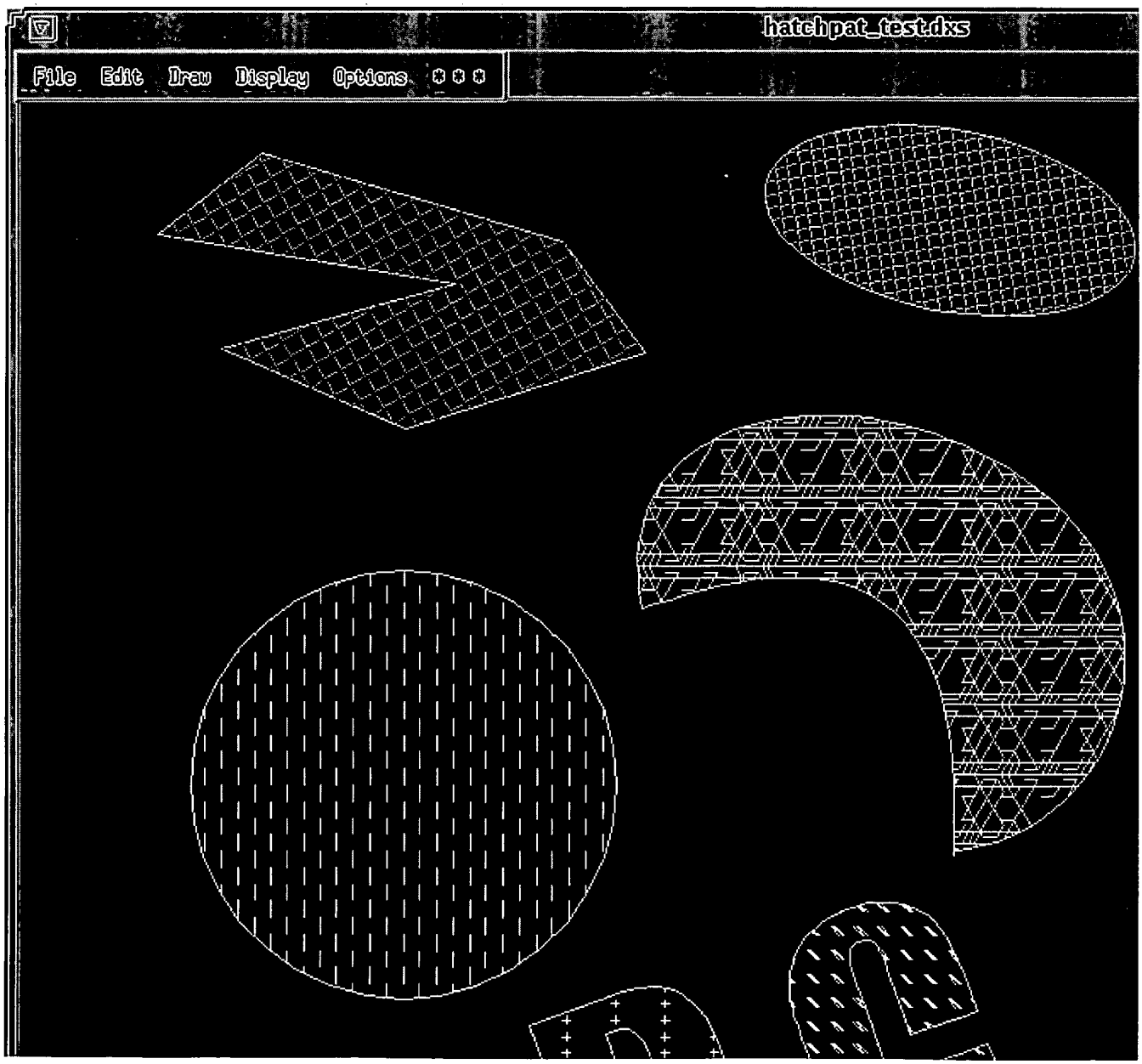
At least 80% of the drafting work people do in AutoCAD may be done in LinuxCAD instead !!!



back to Home Page

Like leading drafting programs LinuxCAD implements various types of hatching, that includes hatching by simple lines and hatching by hatch patterns. You can reuse any and all AutoCAD hatch patterns in LinuxCAD.

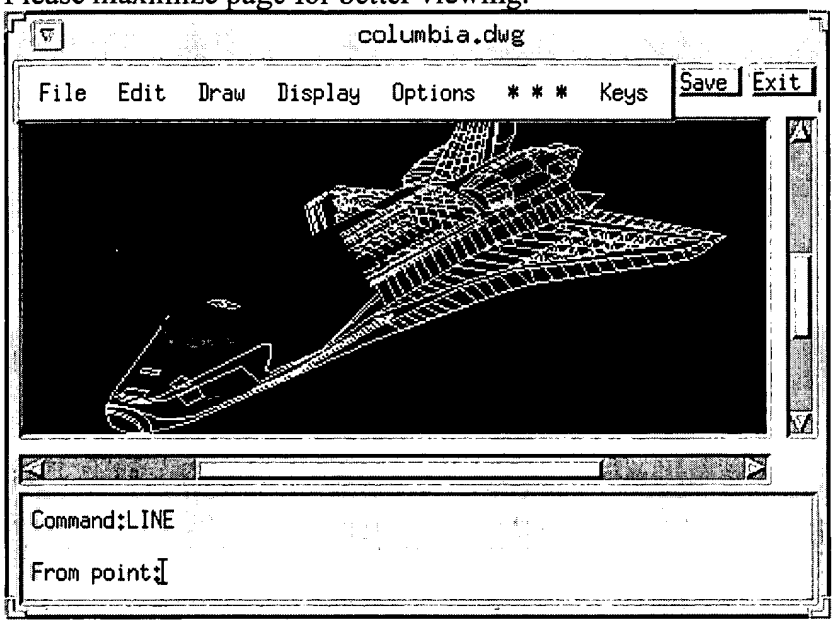
At least 80% of the drafting work people do in AutoCAD may be done in LinuxCAD instead !!!



WWW.LINUXCAD.COM

TEL: 913 663 1724, FAX: 913 661 0120

Please maximize page for better viewing.



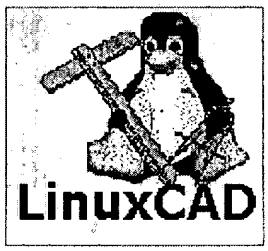
LinuxCAD 2002

Computer Aided Design and Diagramming Software for Linux, MS-Windows, Mac OS X, FreeBSD, SGI IRIX, Macintosh, Solaris. (Replaces AutoCAD,Visio and CorelDraw all in one package!!!) Offer expires on 06/15/2003: Get LinuxCAD 2002 v4.0 for $89.00 on CD-ROM, including delivery!

Subscribe

High Quality Professional 2D drafting and 3D wireframe modelling Computer Aided Design Program. For Architects, Mechanical and Electrical Engineers.

Read about LinuxCAD in German, Portugese, Spanish. Buy LinuxCAD. Hardware requirements. Demo version. F.A.Q. LinuxCAD training. LinuxCAD Review Page. The Truth about LinuxCAD. Slide of the week. More 3D slides.



created in 1995, enchanced and updated in 1996,1997,1998,1999,2000,2001,2002,2003.

General purpose drafting program for every computer user and the Best application program for Linux. Period. Getting familiar with Linux and LinuxCAD is a wise investment for every computer user.

LinuxCAD is today's leading computer aided design and drafting program for The Linux Operating System.

LinuxCAD has grown in sales and functionality since its original introduction in 1998.

LinuxCAD now represents a complete computer aided design program for Linux ... it is a complete replacement for AutoCAD. For any practical purpose, it implements all major features of AutoCAD in such a way that experienced AutoCAD users do not need additional training to start working with LinuxCAD.

Linux users who are new to CAD will find our Users Manual an easy way to learn the CAD trade. There is so much similarity between LinuxCAD and AutoCAD, that while learning LinuxCAD, you will simultaneously learn AutoCAD.

With remarkable ease you will find yourself creating all kinds of technical illustrations, diagrams, sketches and industrial quality drawings.

Suggested Uses: flowcharting,land surveying,mechanical drawing , architectural drafting, entity relationship diagramming ,software engineering diagramming.

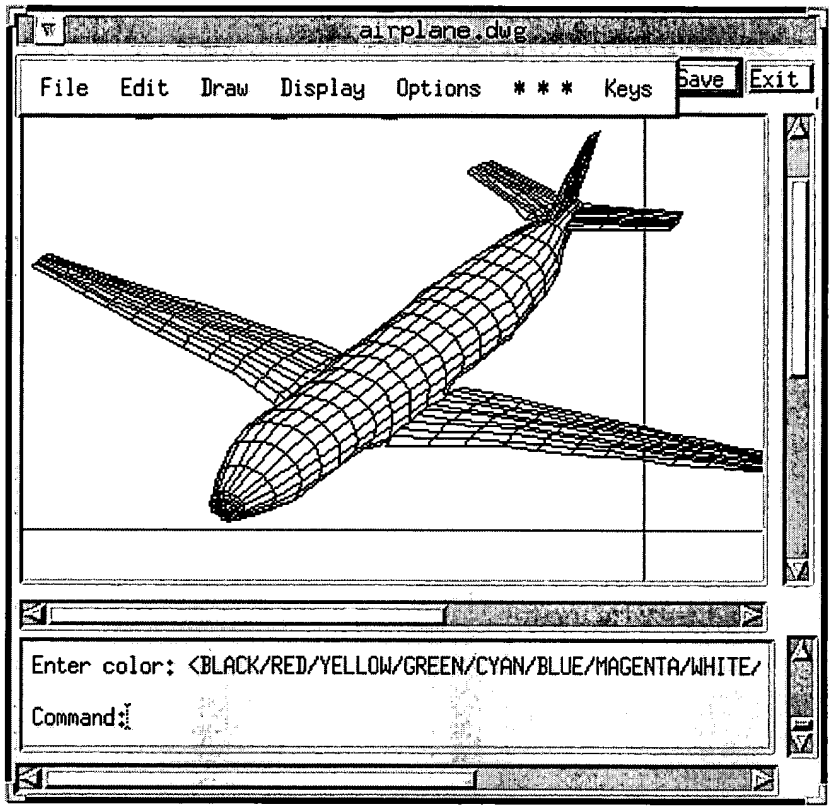
This is a professional quality graphics modelling package for Linux !!!
A must have for every Linux user !!!
Implements all core features of AutoCAD in a very compatible way !!!
No learning curve for AutoCAD users !!!

Download LinuxCAD for $99.00 only click to order !

(if this ordering page does not work with your browser, follow the link "How to purchase" at the bottom of this page)

Special discount for businesses limited time offer

If you already using LinuxCAD, click here.
Developers click here.



LinuxCAD for Multiple Platforms

It should be emphasized that LinuxCAD can run not only on Linux but also on many other Unix platforms. LinuxCAD is a full featured Computer Aided Design System available for Linux on Intel PCs and Sun Microsystems Workstations such as Ultra 5, 10 and others, running Solaris OS. LinuxCAD can also run on LinuxPPC system for Apple Power Macintosh computers.

Here is a list of just a few distinguished features of the

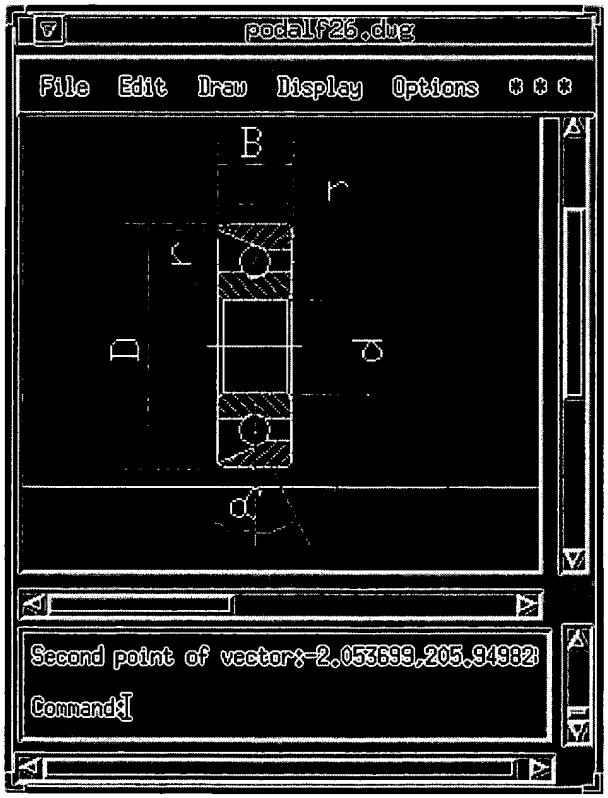
LinuxCAD Computer Aided Design System:

- All commonly used commands of Acad for 2D and 3D drafting, editing and displaying are implemented in LinuxCAD exactly as they are in Acad (including but not limited to: LINE, POLYLINE, CIRCLE, ARC, RECTANGLE, SOLID, POINT, TRACE, DONUT, TEXT, ELLIPSE, TRIM, BREAK, EXTEND, OFFSET, COPY, MOVE, ROTATE, SCALE, MIRROR, INSERT, CHANGE, CHPROP, ARRAY, BLOCK, ATTDEF, EXPLODE, WBLOCK, PURGE, ZOOM, PAN, BREAK, CHAMFER, FILLET, HATCH, STRETCH, ... and many others) they are all there in LinuxCAD. And they work exactly the same way; even the overwhelming majority of messages are the same or almost the same. You may keep using all of the Acad-skills you have. There is no need to relearn !!!
- NEW! Extensive symbol libraries.
- LinuxCAD supports DXF, DWG, DXS, SLD, SHX graphics formats !!!
- sophisticated entity selection mechanism (**Select objects:** with all the standard modes of selection set creation like: Add, Remove, Window, Crossing)
- object oriented entity and object attachment mechanism (includes Nearest, Midpoint, Center, Intersection, Tangent, Perpendicular and others)
- support for fonts in SHX and True Typeformats, all text creation modes of Acad supported, including true fit text creation mode
- NEW! DVIEW command for 3D can change the location of the camera and target in interactive mode. It works very simply: you select the objects and then adjust the observer position. The 3D view in the current viewport will change accordingly.
- allows drawing with virtually any color as a combination of RGB values
- user programmable menus
- EXPORT OF THE DRAWINGS TO POSTSCRIPT either as Black and White or Color and the postscript output can be easily customized using command dialogs and OS environment variables
- PRINTING OF THE DRAWINGS TO MS-Windows Printserver
- variety of drawing entities: lines, circles, traces, solids, arcs, ellipses and polylines
- easy to use GUI command line interface and command scripts
- Axis, Grid and Snap modes
- Ortho mode
- predefined and user programmable line types
- You can load Acad line types from acad.lin straight into LinuxCAD drawing
- LinuxCAD supports cut and paste of graphics CAD models through X-clipboard from one instance of LinuxCAD to another
- allows adjustement of the drafting window size to make the most of your display LinuxCAD can give you whatever big drawing area you want up, to maximum that the X-Window provides in the specified graphics mode
- or you may work on light background instead of black
- you can export drawings you create in LinuxCAD to other CAD systems
- NEW! UCS or User Coordinate System implemented and works, just like in AutoCAD
- NEW! Blocks and attributes fully supported now
- NEW! Linear and circular polylines with line types
- NEW! Up to four viewports to look at 3D model space; you can select different observer locations for every viewport
- NEW! Hidden Lines Removal on 3D objects
- NEW! LinuxCAD now supports Bezier Curves and B-splines which allow for esthetic design of smooth nonrectangular shapes
- user programmable toolbar-like menus and toolbars remember their position on the screen and are extremely convenient when you work with LinuxCAD
- the CAD package allows to work with vector CAD models and with raster scanned pictures

- print preview and autosave (saving of the drawing every minute or other specified time interval)
- highly developed dimensioning subsystem, automatic creation of horizontal, vertical, aligned linear dimensions, radii, diameters, angular dimensions. The LinuxCAD dimensioning subsystem can be fully configured via Linux environment variables and can be easily customized to individual requirements
- integration with databases is possible by means of using Advance LinuxCAD application Development system for C/C++. ADS included in base version !!!
- LinuxCAD is not just a product, it is a product line. Ask about mechanical libraries for LinuxCAD
- NEW! Architectural Library for LinuxCAD now is available instantly, see the order form product selection list for details

All of the above is included in the basic version. The $99.00 includes: 3D, DXF import and export, DWG support. Its all there in new version; extensions are gone; everything is built in !!!

A major advantage is the availability of the package for the industries most reliable and flexible operating systems, LINUX and also most other Unix versions.

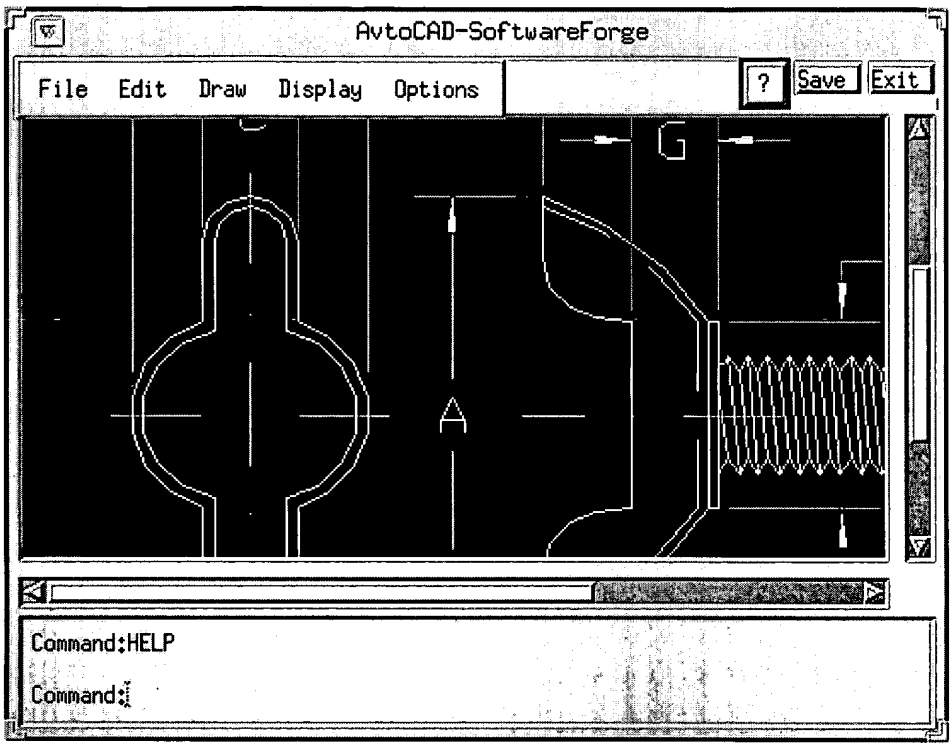


LinuxCAD will enable you to draw almost anything you can imagine. LinuxCAD presents numerous ways in which users can accomplish similar tasks and provides many opportunities in which you can customize and automate your drafting work. All these capabilities are even more enhanced in LinuCAD 3D release 2.0. LinuxCAD is smoothly integrated in the Linux and X window environment. You will be able edit the same file on multiple windows and even multiple displays, copy parts of the drawing using X-clipboard between different instances of LinuxCAD, work on ten on more drawings simultaneously on the same computer.

The LinuxCAD drawing editor has very advanced and extendable internal architecture that allows easy extension with new entity types and retargeting of the package for different application markets, like schematic design, mechanical drafting, block flow chart drawing and other areas. The internal structure of the program and algorithms allows LinuxCAD to utilize parallel processors if they are available. When developing LinuxCAD and implementing DWG support we had a chance to avoid some mistakes that the developers of real AutoCAD made in their earlier releases.

You may think of SoftwareForge Inc., LinuxCAD as about a product or as a technology. Our technology can be applied in consulting services for accomplishing your product delivery objectives. Whatever your field is, engineering , architecture ,

	medical information technology , real time control systems or computer games development, you may benefit from our advanced graphics modeling technology. LinuxCAD is something every Linux user needs, not only drafters or engineers. Software professionals can use LinuxCAD as an Entity Relationship, Flow Chart and Object Visual Modeling tool.



The pricing on all our order forms includes the shipping to any location worldwide! When purchasing LinuxCAD you get more for your money than any other program for use with Linux!

Unlimited use LinuxCAD licenses for business customers, limited time offer, click here
TEMPORARY REDUCED PRICES ON Solaris SPARC/x86 versions of LinuxCAD !!!, click here
Turnkey LinuxCAD solutions and consulting services are available, click here
Linux hardware solutions

LinuxCAD creates a new graphics presentation layer for the Linux system. Many of the software development and system administration work you do now can be organized graphically. LinuxCAD is something every Linux user needs !!!

How to Purchase


e-mail to: sales@softwareforge.com

e-mail link to: SoftwareForge Inc. - developers of the LinuxCAD
Employment Investor relations

Inside AutoCAD 14

Previous Chapter Next Chapter Contents

- 16 -

Text Annotation

---

by Francis Soen

Text is a very important part of any drawing. On any given drawing, you may need to draw a single word, a single sentence, or even paragraphs of text. Being able to efficiently draw and edit text directly affects your productivity. In this chapter you learn how to do the following:

- *Draw and edit single lines of text*
- *Define and use text styles to control the appearance of your text*
- *Draw and edit paragraphs of text*
- *Perform a spelling check on your drawing*
- *Invoke Quick Text mode, mapping fonts, and the clipboard*

Drawing Single-Line Text

A single line of text can consist of a single character, a word, or a complete sentence. The easiest way to draw such text is to use the DTEXT command. To insert a single line of text, from the Draw menu, choose Text, Single Line Text. The initial prompt displayed in the command window presents several options:

```
Justify/Style/<Start point>:
```

The default option is to specify the lower-left endpoint, which is otherwise known as the start point, of the new line of text. After picking the start point, you are prompted to supply the height and rotation angle of the text and the new text to be drawn. As you type the text to be drawn, it is displayed on your drawing. If you make a typographical error, you can use the Backspace key to delete the error and retype the text. You signify the end of the line of text by pressing the Enter key, at which point you may begin a new line of text immediately below the line of text just drawn. To stop drawing any additional lines of text, press the Enter key without typing any new text. You may also relocate the text marker (the []) by picking a point with the cursor.

When you enter text, you can take advantage of the command line buffer to repeat previously entered text by using the up and down arrow keys to scroll through the buffer.

TIP: The spacing between successive lines of text is fixed at approximately 1.67 of the text height. This spacing is normally fixed; however, each line of text is a separate object. As such, you can use the MOVE command to re-arrange the lines. You also can pick a new justification point at the Text: prompt prior to typing the new line of text, thus enabling you to override the default line spacing.

Typing the text you want to draw is the easy part. It is also important to know how to format the text

according to your needs. The following sections discuss how to choose the correct text height, justification, and text style. You also learn how to continue text below the previous line, use special formatting codes and symbols, and edit text.

In the following exercise, you use DTEXT to add several lines of text to a site drawing.

DRAWING SINGLE-LINE TEXT USING DTEXT

1. Open the IAW drawing ACME.DWG from the accompanying CD. This is a site drawing set up in paper space with a model space scale of 1 drawing unit equal to 100 feet (1:100 scale).

2. Make sure you are in paper space. Choose Named Views from the View menu to restore the view TITLE_BLOCK. Make a new layer named **TEXT** and make it current.

3. From the Draw menu, choose Text, Single Line Text. Using the Style option, set the current style to STANDARD. Using the Justify option, set the justification option to Center and pick the point 29,2.5 as the center point. Specify a height of 0.3 and a rotation angle of 0.

Type the text **ACME Engineering** and press the Enter key twice.

4. Repeat the DTEXT command. Pick the point 26.75,2 as the start point. Remember, the default justification is left-justified text. Use a height of 0.2 and a rotation angle of 0.

Type the text **Bakersville Project** and press the Enter key once. Type the text **Legend & General Notes** and press the Enter key twice.

5. Repeat the DTEXT command. Type **M** at the start point prompt, thereby specifying Middle justification.

You can bypass the Justify option and set the justification directly at the initial prompt. Pick the point 29.5,1 as the middle point. Use a height of 0.2 and a rotation angle of 0.

Type the text **L10** and press the Enter key twice.

6. Repeat the DTEXT command. Type **R** at the start point prompt to specify Right justification.

Pick the point 27.8,0.6 as the right point. Use a height of 0.1 and a rotation angle of 0.

Type the text **1"=100'** and press the Enter key twice.

7. Restore the view ALL. Make the model space viewport active (double-click the Paper button).

8. Issue the DTEXT command. Pick the point 855,1290 as the start point.

Use a text height of 50, which at a 1:100 plot scale will produce $^{1}/_{2}$" text. Use a rotation angle of 10 degrees.

Type **Route 101** and press the Enter key twice.

9. The exercise is complete and you may now save the drawing. Figure 16.1 shows the outcome.

Figure 16.1 *ACME.DWG with some single line text added.*

Now that you know how to create new text in your drawing, the following section details how to change text styles in accordance with your needs.

Choosing the Correct Text Height

The hardest part of drawing text is deciding on the correct text height for the scale for which the drawing is set up. Unfortunately, because AutoCAD does not have a built-in mechanism for storing and using the drawing scale to set the correct text height for full-size drawings, it is necessary to take into account the drawing scale in specifying the text height. Use tables 16.1 and 16.2 to help specify the correct text height. To use these tables, go to the row associated with your drawing scale. Then move along the row to the column associated with the height you want your text to have on your plot.

Table 16.1 Text Heights for Architectural Scales

	Plotted Text Heights				
Drawing Scale	*3/32"*	*1/8"*	*3/16"*	*1/4"*	*3/8"*
1/16"=1'	18"	24"	36"	48"	72"
3/32"=1'	12"	16"	24"	32"	48"
1/8"=1'	9"	12"	18"	24"	36"
3/16"=1'	6"	8"	12"	16"	24"
1/4"=1'	4.5"	6"	9"	12"	18"
1/2"=1'	2.25"	3"	4.5"	6"	9"

Table 16.2 Text Heights for Decimal Scales

Drawing Scale	*Plotted Text Heights*				
	3/32"	*1/8"*	*3/16"*	*1/4"*	*3/8"*
1:10	0.9375 d.u.*	1.25 d.u.	1.875 d.u.	2.5 d.u.	3.75 d.u.
1:20	1.8750 d.u.	2.50 d.u.	3.750 d.u.	5.0 d.u.	7.5 d.u.
1:50	4.6875 d.u.	6.25 d.u.	9.375 d.u.	12.5 d.u.	18.75 d.u.
1:100	9.3750 d.u.	12.50 d.u.	18.750 d.u.	25.0 d.u.	37.5 d.u.

**d.u. stands for drafting units*

Choosing a Justification

The default option of DTEXT is to specify the left endpoint, or the start point, of the line of text. Specifying the Style option at the initial DTEXT prompt displays the following prompt:

```
Align/Fit/Center/Middle/Right/TL/TC/TR/ML/MC/MR/BL/BC/BR:
```

Figure 16.2 shows the various justification options and their corresponding locations.

Figure 16.2 *The possible justification points for a line of text.*

Unlike the justification options illustrated in figure 16.2, the Align and Fit options require that you define two points.

Use the Align option when you want to specify the left and right endpoints of the text and do not care about the resulting height. The text height is automatically set to make the text fit between the specified points. Also, the angle from the first point to the second point is used as the rotation angle of the text.

Use the Fit option when you want to specify the left and right endpoints and the height of the text. To make the text fit between the specified points, the height-to- width ratio of the text characters is varied. Therefore, you may end up with skinny-looking characters on one line and very fat-looking characters on the next.

TIP: You can specify a justification option at the primary DTEXT prompt, which eliminates the need for first selecting the Justify option. The most commonly used ones are the default Start point, Right, Middle, and Center justifications.

When the text is initially drawn with one of the alternate justification options specified, it is drawn left justified, as if the default justification were being used. Upon ending the DTEXT command, however, the text is redrawn with the correct justification.

NOTE: You can pick the justification point of a line of text with the INSERT object snap mode.

This concludes the discussion of text justification options. The next section deals with the topic of text styles.

Choosing a Text Style

The appearance of the text drawn by DTEXT is controlled via a named group of settings referred to as a *text style*. The default text style supplied in the prototype drawings ACAD.DWG and ACADISO.DWG is STANDARD. In the template drawings, several text styles are predefined for you. Use the Style option to set the style you want to use to generate the new text. The process of actually defining new and modifying existing styles is discussed in the section "Defining Text Styles," later in this chapter.

Continuing Below the Previous Line

If, after you end the DTEXT command, you want to draw an additional line of text below the last drawn line of text, you can easily do so by issuing the DTEXT command and pressing the Enter key rather than picking a new start point. DTEXT will then draw the new line of text using the style, height, and angle of the previously drawn text.

TIP: To help you spot the last line of text drawn, that line is highlighted when you begin DTEXT. The highlighting, however, may not be apparent if the text is too small on the screen.

Using Special Formatting Codes and Symbols

You can do a limited amount of formatting with the DTEXT command. For instance, a line can be drawn under or above the text simply by adding the codes %%u (underlining) and %%o (overlining) to the text as you enter it. The codes act as toggle switches; the first time you include the code in a line of text, it turns that effect on. The second time the code is encountered in the same line of text, the effect is turned off. If the code is not encountered a second time in the line of text, then the effect is continued to the end of the text line but is not continued to the next line. For example, to draw the text shown in figure 16.3, the text you type is **%%uUnderlining%%u and %%oOverlining%%o can be used separately or %%o%%utogether**.

Figure 16.3 *Using underline and overline formatting codes.*

In addition to underlining and overlining, you also can draw symbols that are in the font file, but are not on the keyboard. Table 16.3 shows several formatting codes and the resulting symbols.

Table 16.3

Additional Formatting Codes

Formatting Code	*Symbol*	*Meaning*
%%c	Ø	diameter
%%d	[ring]	degree
%%p	±	plus/minus

The codes are not case-sensitive. In addition to the codes in table 16.3, the code %%nnn can be used to draw any character in a font file.

A much easier way of drawing a symbol is simply to use Window's Character Map program.

To use the Character Map program in place of the %%nnn code, simply start the Character Map program (usually found in the Accessories group of programs) and select the font file you have specified in the current text style. Then select the character you want to draw and copy it to the clipboard. You can now paste the character into the text you are typing.

WARNING: Not all font files contain the same characters, which is why it is important that the font file you choose to copy from in the Character Map program is the same font file specified in the text style you are drawing with in AutoCAD. What the Character Map program actually copies when you choose to copy a character to the clipboard is the character's position number in the font chart. When you paste that character in AutoCAD, the character corresponding to the position number recorded in the clipboard is drawn--if you are using a different font file in AutoCAD, you may end up with a different character

altogether.

After drawing and formatting the initial text, you may want to change the wording or appearance of the text. The following section covers the commands you will need to do this.

Editing Single-Line Text

Two commands are of particular use for editing existing text: DDEDIT and DDMODIFY. DDEDIT is quicker to use than DDMODIFY when all you want to do is change the text in one or more text objects. DDMODIFY is slower, but more powerful, than DDEDIT, in that DDMODIFY enables you to change the appearance of the selected text.

Using DDEDIT

From the Modify menu, choose Object, Text to issue the DDEDIT command. After selecting the text object to be changed, the Edit Text dialog box appears displaying the selected text (see fig. 16.4).

Figure 16.4 *The Edit Text dialog box of DDEDIT.*

Initially, the entire line of text is highlighted and will be replaced by whatever you type. If you want to edit a specific portion of the text, it is necessary to position the cursor at the desired point in the text and pick it. You can then use the Insert, Delete, and Backspace keys to add and delete characters.

If you want to replace a portion of the text displayed in the Edit Text dialog box, highlight the portion to be replaced. The highlighted portion is replaced with any new text you type.

Using DDMODIFY

Choose Properties from the Standard toolbar or Modify menu and select a single text object to issue the DDMODIFY command. The resulting Modify Text dialog box, shown in figure 16.5, enables you to change the text, the style, the justification point, and the various settings that control the appearance of the text object.

Figure 16.5 *The Modify Text dialog box of DDMODIFY.*

See "Defining Text Styles" later in this chapter for a clearer explanation of the text settings that you can change.

In the following exercise, you use DDEDIT and DDMODIFY to change some text in a drawing.

EDITING SINGLE-LINE TEXT WITH DDEDIT AND DDMODIFY

1. Continue to use the drawing ACME.DWG to which you added some text in the previous exercise.

2. Choose Named Views from the View menu. Restore the view TITLE_BLOCK.

3. From the Modify menu, choose Object, Text. Select the text "Bakersville Project." Highlight "ville" and type **field**. The text should look like figure 16.6. Click on OK to make the change. Press Enter to exit the DDEDIT command.

4. Choose Properties from the Object Properties toolbar. Select the text "L100." Press the Enter key to end the selection process. Change the height from 0.20 to 0.30 and click the OK button to effect the height change. The text should look like figure 16.6.

5. You are done making changes to this drawing, so you may now save the drawing.

Figure 16.6 *Changing text with DDEDIT and DDMODIFY.*

Defining Text Styles

A *text style* is a named group of settings that controls the appearance of text in a drawing.

The default text style, and only defined style, in the templates ACAD.DWT and ACADISO.DWT is named STANDARD. You can, however, have as many text styles defined as you want in a drawing (the other template files each have several styles predefined). Text styles are defined and modified with the STYLE command, which is issued by choosing Text Style from the Format menu. Figure 16.7 shows the Text Style dialog box. The various settings within the Text Style dialog box are explained in more detail in the following sections.

Figure 16.7 *The Text Style dialog box of STYLE.*

To create a new style, you actually begin by making a copy of the current style. If the current style is not the style you want to begin with, then select the desired style from the list of existing styles (thereby making it the current style).

Click on the New button. Specify a name for the new style, and a duplicate style is created from the selected style. To rename an existing style, select the style from the list of existing styles, click on the Rename button, and enter a new name. To delete an existing style, highlight the name from the list of existing styles, and click on the Delete button. The Standard text style cannot be renamed or deleted.

NOTE: When a text object is created, the style it is created with is recorded with the object. A text style can only be deleted if no existing text objects reference the style.

Text styles are stored in the drawing in which they are defined. If you want to have multiple styles immediately available in a new drawing, define the styles in your template drawings. If you want to import a style from another drawing, insert the other drawing or attach the drawing as an external reference and bind the desired style. (See Chapter 13, "External References," for a complete explanation of binding.)

In defining a new style or modifying an existing style, you must choose a font file, what special effects to enable, a text height, a width factor, and an oblique angle. Choosing these settings and previewing the results of these settings are covered in the following sections.

Previewing the Text Style Settings

The character Preview area enables you to view a sample of the selected style and the results of changing the various settings. To view your own sample text, type your sample text in the text edit box and click on the Preview button.

Choosing a Font and Style

The font file is the file that contains the information that determines the shape of each character. Table 16.4 lists the various types of font files supplied with AutoCAD.

Table 16.4 Various Types of Font Files

File Name Extension	*Font Type*
SHX	AutoCAD's native font file, known as a shape file
TTF	TrueType font file

The most efficient font files are the shape files supplied with AutoCAD. In addition to the TrueType font files supplied with AutoCAD, the TrueType fonts supplied with Windows and other Windows applications can also be used.

PostScript files used to be directly usable in earlier versions of AutoCAD. In R14, however, you must first use the COMPILE command to compile the PostScript font file into a shape file. In fact, in earlier versions several PostScript font files were supplied, but none are supplied in R14.

AutoCAD supports TrueType font families, which means that for some TrueType fonts, you can choose a font style such as regular, italic, bold, or bold italic. Note that not all TrueType fonts have more than the regular style defined.

There are two system variables that affect the plotting of text drawn with TrueType fonts, TEXTFILL and TEXTQLTY. When TEXTFILL is disabled, the characters are plotted in outline form only. If TEXTFILL is enabled, the characters are filled in.

The value of TEXTQLTY affects the smoothness of the characters at plot time. The value of TEXTQLTY can be set from 0 to 100, with the default value set to 50. The higher the value, the better the resolution of the characters, but it will take longer to process the drawing for plotting.

Both system variables can be typed at the Command: prompt, but are easier to access through the Print/Plot Configuration dialog box (see fig. 16.8). TEXTQLTY is set via the Text Resolution text edit box, while TEXTFILL is set via the Text Fill check box.

Figure 16.8 *Setting TEXTFILL and TEXTQLTY through the Print/Plot Configuration dialog box.*

TIP: Using the simplest shaped characters will minimize the drawing size and speed up opening and working with the drawing file. The characters in the Simplex and Romans font files are quite simple in appearance and are similar to the simplex characters used in board drafting. Some shape files contain the alphabet of foreign languages, such as GREEKS.SHX, or even symbols, such as SYMUSIC.SHX.

After you change the font file associated with an existing style, upon applying the change, all text drawn with the modified style is updated to reflect the change. If you want to draw text with more than one font file, you must create one style per font file and switch between the styles as you draw the text.

Setting a Height

Also found in the Font area is the text Height setting. The default height of 0 dictates that the user sets the text height at the time the text is drawn. A height other than 0 sets the text height for that particular style to that height. The style is then referred to as a fixed height style and the text height prompt for the DTEXT command is suppressed.

Changing the text height setting of an existing style does not affect the appearance of existing text objects.

Specifying Special Effects

In the Effects section of the Text Style dialog box are the Upside down, Backwards, Vertical, Width Factor, and Oblique Angle settings. These settings are covered in detail in the following sections.

Upside Down, Backwards, and Vertical Text

In the Effects area, you can enable the Upside down, Backwards, and Vertical settings. See figure 16.9 for an example of how these settings affect the appearance of text.

Figure 16.9 *The effects of Upside down, Backwards, and Vertical settings on text.*

Although the Upside down and Backwards options work with all font files, the Vertical setting only works with SHX files.

TIP: If you want to draw text upside down, you don't have to enable the Upside down option. Instead, specify a text rotation angle of 180 degrees. The Backwards option is useful if you want to plot text on the backside of the plot sheet so that the text is readable when viewed from the front. The Vertical option is useful when you need to draw text down the side of a vertical surface, such as a building.

Unlike the font file setting, changing the Upside down and Backwards settings of an existing style does not result in the existing text being automatically updated to reflect the changes. Changing the Vertical setting, however, does affect existing text objects, so you may want to create a new style before changing the Vertical setting.

Setting a Width Factor

The Width Factor determines the width-to-height ratio of the drawn characters. A factor of 1 results in the characters being drawn with the width-to-height ratio defined in the font file used. A factor greater than 1 results in fatter characters, while a factor less than 1 results in skinnier characters. Figure 16.10 illustrates the effects of using different width factors. All three lines were drawn with the same text height.

Figure 16.10 *The effects of the Width Factor setting on a line of text.*

TIP: Drawing text with a width factor that is less than 1 may make it easier to fit text into

an already crowded drawing.

Setting an Oblique Angle

The Oblique Angle setting affects the slant of the characters. It is often used to draw italic text when the characters in the font file being used are not naturally italic. Unlike the text rotation angle, the oblique angle of 0 refers to a vertical direction (see fig. 16.11). A positive text value makes the letters lean to the right, and a negative value makes the letters lean to the left.

Figure 16.11 *The effects of the Oblique Angle setting on a line of text.*

In the following exercise, you use the STYLE command to modify an existing style and to create a new style.

MODIFYING AND CREATING TEXT STYLES

1. Continue to use the drawing ACME.DWG. Restore the view TITLE_BLOCK.

2. Issue the STYLE command by choosing Text Style from the Format menu. In the Text Style dialog box, make sure STANDARD is the current style.

3. Select the ROMANS.SHX font file. Click on the Apply button and close the dialog box. Notice how the text you drew in the previous exercise, with the Standard text style, is revised to reflect the font file change (see fig. 16.12).

Figure 16.12 *Modifying the appearance of text by changing the font file in the text style definition.*

4. Repeat the STYLE command. Click on the New button. Name the new style **NOTES**. Initially, NOTES is a duplicate of STANDARD, the style that was current at the time the New button was selected.

5. Choose the TrueType font Courier New, which is a font file that offers several font styles. If you do not have Courier New on your list of available fonts, then choose an alternate font. Choose Bold as the Font Style. Click on the Apply button and close the dialog box. The text style NOTES is now the current text style.

6. After completing the exercise, save the drawing.

Drawing Paragraphs of Text with MTEXT

While the DTEXT command can be used to draw multiple lines of text, each line is drawn as a separate object. Sometimes you will want to draw multiple lines of text as a single unit, such as a paragraph of text. At such times, use the MTEXT command (see fig. 16.13), which is issued by choosing Text from the Draw toolbar or Multiline Text from the Text submenu of the Draw menu.

Figure 16.13 *The Multiline Text Editor of MTEXT.*

After you issue the MTEXT command, you are prompted to select the first corner point of a window.

This window is used to determine the direction in which the mtext object is drawn. When the window is dragged to the right, the mtext object is drawn to the right; when the window is dragged to the left, the mtext object is drawn to the left. Similarly, when the window is dragged upward, the mtext object is drawn upward; when the window is dragged downward, the mtext object is drawn downward. Within the window, the mtext object is drawn with a top left justification. If you want, you can change the justification type to one of eight others: TC (Top Center), TR (Top Right), ML (Middle Left), MC (Middle Center), MR (Middle Right), BL (Bottom Left), BC (Bottom Center), or BR (Bottom Right). These justification types are similar to those available with the DTEXT command (refer to fig. 16.2), except that they apply to the whole mtext object and not just a single line of text.

If you want, you can also choose to use the first window point as the justification point by specifying the Justify option and choosing a justification option. Then choose the Width option and enter the width rather than using the width of the window as the width.

A width of zero will disable the word wrap feature of the Multiline Text Editor, and you will have to press the Enter key every time you want to begin a new line of text.

Several other options appear at the command line that can be set from the command line, but which are easier to set through the Multiline Text Editor dialog box. The Multiline Text Editor is divided into two parts. The bottom part is the screen editor and the top part is divided into three tabs: the Character tab, Properties tab, and Find/Replace tab, all of which are described in detail in the following sections.

TIP: If you position your pointer in the screen pointing area and right-click on your mouse, a menu appears that facilitates access to the Undo, Cut, Copy, Paste, and Select ALL operations.

If you have text in an existing ASCII or RTF file, use the Import Text button to import the file into the editor and then edit the text as you want.

Using the Character Tab

The Character tab controls the properties of the text you are drawing. The property settings can be used in one of two ways. First, the property settings control the appearance of the text you type. You also can change the properties of selected text through these settings, thereby creating various special effects. To select text, position the cursor at the beginning of the text, left-click, and then drag the cursor to the end of the text. You can select a word by double-clicking at the start of the word or select the entire body of text by triple-clicking.

Using Special Effects

This section discusses the various effects you can achieve by using the Character tab of the Multiline Text Editor dialog box.

Changing Font File and Text Height

After selecting the text to be affected, you can change the font file to be used and even the height of the text. The text height drop-down list is actually a combination drop-down list and text edit box. You can enter a new text height in the text edit box, or you can also select a height that was previously entered

from the drop-down list.

Bold and Italic Text

The Bold and Italic buttons enable you to bold or italicize the text, but only if the chosen font file is a TrueType font. You can use the Underline button to underline any selected text regardless of the font file used. To remove any bold or italic effects, simply select the text again and choose the appropriate button.

Stacking/Unstacking Text

The Stack/Unstack button is used to stack or unstack the selected text. Selected text can be stacked when a backslash character (/) appears somewhere in the text. Everything to the left of the slash is treated as the numerator and everything to the right is treated as the denominator.

Color Settings

Normally, the color setting is ByLayer, but if you want, you can set a specific color for the selected text with the color drop-down list. Just remember that Text Color controls the pen used with the PLOT command.

Using Special Symbols

Use the Symbol drop-down list to insert the degree, plus/minus, or diameter symbol (see fig. 16.14). To insert any other symbol, choose Other from the list to invoke the Character Map program. Inserting a nonbreaking space prevents the Multiline Text Editor from making a break at that point when deciding where to break the line of text (word wrap feature) to continue to the next line.

Figure 16.14 *The Symbol drop-down list in the Multiline Text Editor.*

Using the Properties Tab

Choosing the Properties tab enables you to set the text style, justification option, width, and rotation angle of the overall mtext object. Remember that if you use a window to define the location of the mtext object, the justification used is TL, or Top Left, and that the width of the window is the width used for the mtext object.

Using the Find/Replace Tab

Use the Find/Replace tab to search for a specific combination of characters and even to replace the found text with a replacement text string. If the Match Case setting is enabled, only text that matches the case of the find string exactly is found. If the Whole Word setting is enabled, only words that exactly match the find string are found; otherwise, even words that simply contain the find string are found.

After specifying the settings, use the Find button to start the search.

Editing Mtext Objects

Edit mtext objects with the DDEDIT and DDMODIFY commands as you would edit text objects. If you

want to change a property, such as height, of the mtext object, you must select the entire body of text and then change the property. Don't forget, a triple-click selects all text.

Additionally, you can use grips to move or change the width of the mtext object. When you select the grip point that corresponds to the justification point, the mtext object can be moved. Selecting any other grip point enables you to stretch the width of the mtext object.

The next exercise uses mtext to add the notes shown in figure 16.15.

USING MTEXT TO DRAW PARAGRAPHS OF TEXT

1. Continue to use the drawing ACME.DWG. Restore the view ALL. Make sure the paper space viewport is active and the layer TEXT is current.

2. Issue the MTEXT command by choosing the Text tool from the Draw toolbar.

Specify the point 26,18 as the first corner point. Enter **@4,-1** for the opposite corner point. The Multiline Text Editor dialog box is displayed. Set the height to **0.25**. Enter the text **Notes** and press the Enter key twice.

3. Enter the text. The information on this drawing reflects information gathered as of 2/2/97. Press the Enter key twice.

4. Enter the text. This drawing is a preliminary drawing and should not be used for engineering purposes.

5. Select the text Notes and click on the Underline button. Close the dialog box.

Figure 16.15 shows the drawing with the MTEXT object added.

Figure 16.15 *ACME with an mtext object added.*

6. For now, you are done modifying this drawing, so go ahead and save your work.

Performing a Spelling Check

To check the spelling of your text and mtext objects, issue the SPELL command by choosing the Spelling tool from the Standard toolbar. Figure 16.16 illustrates the Check Spelling dialog box of SPELL.

Figure 16.16 *The Check Spelling dialog box of SPELL.*

When SPELL encounters an unknown word, the Check Spelling dialog box is displayed, and you must choose to either replace the word, ignore the discrepancy, or add the word to your supplemental dictionary. If no errors are found, a message box appears informing you that the spell check is complete, but the Check Spelling dialog box itself does not appear.

The following exercise takes you through the steps of using the SPELL command to check the text you entered in ACME.DWG.

CHECKING THE SPELLING IN YOUR DRAWING

1. Continue to use the drawing ACME.DWG. Make sure the paper space viewport is current.

2. Issue the SPELL command by choosing SPELLER from the Standard toolbar.

Use the All option to select all objects. SPELL stops at any word it does not recognize. If you use abbreviations a lot, be sure to add them to your supplemental dictionary.

3. If you want to check the spelling of text in model space, you must repeat the SPELL command with the model space viewport current.

4. This is the last exercise for this chapter, so you may end the drawing if you want.

Specifying the Dictionaries

The SPELL command looks up words in as many as two dictionaries at any given time: a main and a supplemental dictionary. Several main dictionaries are supplied with AutoCAD, with the default being the American English Dictionary. The default supplemental dictionary is SAMPLE.CUS (SAMPLE.CUS contains a number of AutoCAD command words and terms). To change the dictionaries used by SPELL, issue the PREFERENCES command. In the preferences dialog box, change the Main Dictionary and Custom Dictionary File settings under Text Editor, Dictionary, and Font File Names in the Files tab.

Unlike the supplemental dictionary, the main dictionary file cannot be modified or added to. You can, however, add words to and change the supplemental dictionary.

Creating a Supplemental Dictionary

A *supplemental* dictionary file is a simple text file that contains the additional words that you want SPELL to use. The file format is simple--one word per line. You can create as many supplemental dictionaries as you want, but you can only use one at any given time. When you create a supplemental dictionary, be sure to use a CUS file name extension and place it in one of the folders listed in the Support Files Search Path setting in the Preferences dialog box.

Now that you know how to draw single and multiple lines of text, and how to edit the text style and check spelling, you should familiarize yourself with several additional text handling features that are covered in the next section.

Looking at Additional Text Options

The following sections cover several optional text handling features that may prove useful to you. These features will enable you to speed up the display of text, handle missing font files, and insert text files into the current drawing.

Enabling the Quick Text Display

Displaying text, especially text drawn with complex font files, can be time- consuming. When you want to speed up the display of the drawing and do not need to read the existing text, enable the Quick Text

mode. You can enable the Quick Text mode through the Drawing Aids dialog box. With Quick Text enabled, text and mtext objects are displayed as simple rectangles. To immediately see the effects of enabling Quick Text on existing text, issue the REGEN command.

NOTE: Even with Quick Text enabled, new text objects are displayed completely, rather than as rectangles, making adding text easier.

Specifying an Alternate Font File

Font files are not stored with the drawing file. If a font file that is referenced in the drawing is not available when the drawing is opened, an error message is displayed. You are then prompted to choose a replacement font file. If you want to bypass all such error messages, you can specify a font file that is automatically used whenever a needed font file cannot be found. This *alternate font file* is specified by setting the Alternate Font File setting under Text Editor, Dictionary, and Font File Names in the Files tab of the Preferences dialog box. The default alternate font is simplex.shx.

WARNING: A couple of possible problems with using an alternate font occur, however. If the missing font file contains special characters that the alternate font file does not have, the text on the drawing may end up incomplete. Furthermore, because the space that a line of text occupies is dependent on the font file used to generate the text, you may find that the text on the drawing looks out of place or does not fit properly anymore. The best solution is to obtain the correct font files and use them, unless you are sure you have a suitable replacement file.

Mapping Fonts

If you need to specify more than one alternate font file, specify a font mapping file. A *font mapping file* is a text file where each line in the file specifies the font file to be replaced and its substitute font file (separated by a semicolon). The default font map file is ACAD.FMP. You can change the font map used by changing the Font Mapping File setting under Text Editor, Dictionary, and Font File Names in the Files tab of the Preferences dialog box.

The default file ACAD.FMP (the contents of which are listed in the following list) maps the PostScript fonts that were supplied with R13 to their equivalent TrueType fonts.

Cibt__.pfb = CITYB__.TTF

cobt__.pfb = COUNB__.TTF

eur__.pfb = EURR__.TTF

euro__.pfb = EURRO__.TTF

par__.pfb = PANROMAN__.TTF

rom__.pfb = ROMANTIC__.TTF

romb__.pfb = ROMAB__.TTF

romi__.pfb = ROMAI__.TTF

sas__.pfb = SANSS__.TTF

sasb__.pfb = SANSSB__.TTF

sasbo__.pfb = SANSSBO__.TTF

saso__.pfb = SANSSO__.TTF

suf__.pfb = SUPEF__.TTF

te__.pfb = TECHNIC__.TTF

teb__.pfb = TECHB__.TTF

tel__.pfb = TECHL__.TTF

Drawing Text as Attributes

An alternate method to drawing text objects that are to be incorporated into block definitions is to draw attributes. Attributes behave much like text objects but have additional functions beyond displaying text. Attributes are discussed in more detail in Chapter 12, "Creating and Using Blocks."

Dragging and Dropping Text Files

In Windows 95 and NT, you can drag a text file icon from the desktop and drop it into your drawing. AutoCAD will automatically draw the file contents as an mtext object, using the current text settings for the text height, rotation angle, and text style.

Copying Text Using the Clipboard

You also can copy text from any application to your clipboard and then paste the contents into your drawing. If you use the PASTE command, the contents are dropped into your drawing as an embedded object. If you use the PASTESPEC command, you can choose to paste the clipboard contents as text, in which case the text is drawn as an mtext object.

Be prepared to perform some experimenting if you are going to use the clipboard. The clipboard operations depend on OLE (Object Linking and Embedding), which is a constantly evolving feature. Some windows applications support older versions of OLE and may limit what operations you can carry out and how much information you can copy to the clipboard.

Another consideration is that the form in which the data is pasted into your drawing may limit how much information can be pasted from the clipboard. For example, I copied three pages of text from WordPerfect 7.0 onto the clipboard. I then pasted the data into an AutoCAD drawing, but only one page of the text was displayed. But by using PASTESPEC and specifying that the data should be pasted as text, all three pages appeared in the drawing (of course, I lost all the WordPerfect formatting).

Creating Your Own Shape File

You have the option of creating your own shape file containing the characters you want to use. Creating the instructions that define each character is a laborious procedure because you have to break each character into a series of strokes and enter the codes for those strokes into the new font file. In the earlier versions of AutoCAD that did not support the use of TrueType fonts, defining your own shape file was the only way to add to the font files supplied with AutoCAD. With R14 and the support of TrueType font families, it is much more sensible to simply use one of the fonts supplied with Windows. You can also purchase additional fonts from a number of software vendors at a very low cost.

Using the Bonus Text Routines

In the Text submenu of the Bonus menu and on the Bonus Text Tools toolbar, you will find several powerful routines for drawing and editing text (if you do not see the Bonus menu or toolbar, then follow the installation instructions in the "Using Bonus Tools" section of Chapter 11, "Advanced Geometry Editing"). The tools enable you to create some very interesting effects with text and are covered in the following sections.

Adjusting the Width Factor with TEXTFIT

TEXTFIT is issued by choosing Text Fit from the Text submenu of the Bonus menu or from the Bonus Tools toolbar. It is designed to work with text objects only and will not operate with mtext objects. You use TEXTFIT to adjust the width factor of the characters in a text object. After selecting the text object, the following prompt is displayed and a rubber band is displayed:

```
Select Text to stretch/shrink:
Starting Point/<Pick new ending point>:
```

The rubber band is anchored at the left endpoint of the text object. The length of the rubber band represents the distance the line of text is to occupy. The default option is to pick the endpoint of the rubber band, to define that distance. TEXTFIT then calculates and applies a width factor to the text object that will stretch or shrink the line of text to fit the specified distance. Specifying the Starting Point option enables you to move the text to a point you select and then adjust the width factor.

If the selected text was drawn with any justification other than left justification, the point you pick as the endpoint of the rubber band is also used as the new justification point for the text object.

WARNING: If you have to undo the TEXTFIT command, be careful. TEXTFIT changes the UCS (User Coordinate System) during its operation. Unfortunately, issuing a single U will not undo the effects of TEXTFIT, so you will have to issue several U commands to undo the entire command. If you do not issue enough U commands, you may end up restoring the UCS implemented by TEXTFIT, so always check your UCS when undoing TEXTFIT. Instead of using TEXTFIT, you can use the DDMODIFY command to change the width factor of a selected text object and not have to worry about the status of the UCS.

Creating a Mask with TEXTMASK

TEXTMASK is issued by choosing Text Mask from the Text submenu of the Bonus menu or from the Bonus Tools toolbar. It is designed to work with text objects and will not operate with mtext objects. You use TEXTMASK to create a clear area around the text object. Figure 16.17 illustrates a text object with two lines drawn through the text.

Figure 16.17 *Using TEXTMASK to obscure the objects running through a text object.*

By using TEXTMASK, you can create a clear zone around a text object so that the lines do not obscure the text object. The result of using TEXTMASK illustrated in figure 16.17 suggests that the lines were trimmed. In fact, the lines are still intact. TEXTMASK creates a new type of object named *wipeout*. A wipeout object acts as a barrier that obscures objects that it covers. TEXTFIT draws the wipeout object in such a way as to obscure all objects except the text object. After selecting the text object, the following prompt is displayed:

```
Enter offset factor relative to text height <0.35>:
```

The offset factor defines the size of the wipeout object and is a factor that is multiplied with the text height to obtain the final offset distance (see fig. 16.18).

Figure 16.18 *The offset used to size the wipeout object relative to the text object.*

The wipeout object is completely invisible, and the rectangular boundary shown in figure 16.18 is not actually visible. Because a wipeout object is completely invisible, TEXTMASK automatically creates a group to link the wipeout object created with the text object for which it is created. So long as the Object Grouping setting in the Object Selection Settings dialog box is enabled, the invisible wipeout object is automatically selected whenever you select the text object for any editing command.

WARNING: In order for the wipeout object to function correctly, the system variable SORTENTS must be set to 127, which TEXTMASK automatically does. Changing the value of SORTENTS can disable the correct functioning of the wipeout object. SORTENTS is set by choosing the Object Sort Method button of the Object Selection Settings dialog box.

Changing Text with CHT

CHT is issued by choosing Change Text from the Text submenu of the Bonus menu or Change Multiple Text Items from the Bonus Tools toolbar. It is designed to work with both text and mtext objects. With CHT, you can change the following parameters of the selected text:

- The text height
- The text justification
- The text location
- The text rotation angle
- The text style with which the text is drawn

- The text itself
- The width factor of a text object or the width of a mtext object.

You can use DDMODIFY to change the same parameters that CHT affects; however, the advantage of using CHT is that you can select multiple objects, whereas DDMODIFY can only be used on one object at a time.

WARNING: There seems to be a bug with CHT's Location option (enables you to move the selected object) when an mtext object is selected. With a text object, specifying the Location option enables you to move the object. With an mtext object, the Location option has no effect.

Exploding Text with TXTEXP

TXTEXP is issued by choosing Explode Text from the Text submenu of the Bonus menu or from the Bonus Tools toolbar. It is designed to work with both text and mtext objects. TXTEXP replaces the selected text with a group of polylines. Each character in the text is replaced with a polyline that follows the shape of the original character. The end result of using TXTEXP on a text or mtext object is a group of polylines that resemble the original text characters.

Drawing Text Along an Arc with ARCTEXT

ARCTEXT is issued by choosing Arc Aligned Text from the Text submenu of the Bonus menu or from the Bonus Tools toolbar. It is used to draw text along the outside or inside of an arc (see fig. 16.19).

Figure 16.19 *Text drawn along the outside and inside of an arc with ARCTEXT.*

ARCTEXT generates a new type of object, the arctext object. After issuing ARCTEXT, you are prompted to select either an arc or an existing arctext object. Select an arc if you want to draw a new arctext object, and select an existing arctext object if you want to edit the object. After selecting the arc or arctext object, the ArcAlignedText Workshop dialog box is displayed (see fig. 16.20).

Figure 16.20 *The ArcAlginedText Workshop dialog box of ARCTEXT.*

You set the parameters that control the generation of the arctext object with the Format pull-down menu or with the buttons that run along the top of the dialog box. The usage of the controls labeled 1 through p in figure 16.20 is described in the following sections.

Drawing Reverse Text

If you press button 1, the text is drawn backward. This option is similar to the Backwards setting in the Text Style dialog box. The button next to 1, button 2, is inactive as of the time this book was written. Pressing button 2 has no effect whatsoever.

Controlling the Alignment

Buttons 3 through 6 control how the arctext is placed relative to the selected arc. If button 3 is selected,

then the arctext object is positioned relative to the left endpoint of the arc. The offset from the left endpoint is set through the Offset from left text edit box.

If button 4 is selected, then the arctext object is positioned relative to the right endpoint of the arc. The offset from the right endpoint is set through the Offset from right text edit box.

If button 5 is selected, then the arctext object is positioned relative to the midpoint of the arc.

If button 6 is selected, then the arctext object is positioned relative to both the left and right end points. The offset from the endpoints is set through the Offset from left and Offset from right text edit boxes.

Positioning the Text

You use buttons 7 and 8 to position the text along the outside 7 or the inside 8 of the arc. Refer to figure 16.19 for an example of both positions.

Setting the Character Direction

You use buttons 9 and j to choose whether the characters are drawn away from the center of the arc 9 or toward the center of the arc j. Figure 16.21 illustrates arctext objects drawn in both directions.

Figure 16.21 *Arctext drawn toward and away from the center of the arc.*

Setting the Typeface

The Bold k, Italic L, and Underline m buttons are used to set the typeface of the characters and whether the arctext is underlined.

Setting the Color

The color drop-down list (control n) is used to set the color of the arctext. The default color is ByLayer.

Setting the Text Style

The Text Style drop-down list (control o) does not seem to have any effect on an arctext object. You set the font file with the font drop-down list (control p). The text height and width factor of the characters is set with the Text height and Width factor text edit boxes.

Performing a Search and Replace with FIND

FIND is issued by choosing Find and Replace Text from the Text submenu of the Bonus menu or from the Bonus Tools toolbar. It is designed to work with text objects only and will not operate with mtext objects. Use FIND when you want to replace one text string with a new text string. After FIND is the Find and Replace dialog box is displayed (see fig. 16.22).

Figure 16.22 *The Find and Replace dialog box of FIND.*

Type the text you are searching for in the Find text edit box. Type the text you want to replace the found text with in the Replace With text edit box. If you want to have the search be case-sensitive, then enable the Case Sensitive setting. If you want to automatically search all text objects in the drawing, then

enable the Global Change setting. If the Global Change setting is not enabled, then you will be prompted to select the text objects to be searched.

After you set the parameters for the search, click on the OK button. When the first occurrence of the search string is found, a new Find and Replace dialog box is displayed (see fig. 16.23).

Figure 16.23 *The Find and Replace dialog box is displayed when the specified text is found.*

The number of text objects that have been found to contain the specified text is displayed in the lower-left corner of the dialog box. Click on the Replace button if you want to replace the found text with the replacement text. Click on the Skip button if you do not want to replace the found text. Click on Auto if you want all occurrences of the found text to be automatically replaced.

Summary

AutoCAD provides a variety of tools to deal with drawing and editing text. This chapter covered the basic steps needed to deal with single lines of text in your drawings, as well as how to add multiple paragraphs of text using the MTEXT command. Editing text and defining and changing text style to control the appearance of text was also covered. But remember, using the simple font files such as SIMPLEX.SHX will make text-rich drawings easier to deal with in the long run.

← Previous Chapter → Next Chapter ↑ Contents

Inside AutoCAD 14

← Previous Chapter → Next Chapter ↑ Contents

- 17 -

Drawing Hatch Patterns

by Francis Soen

When you need to fill an area with a repetitive pattern, you can use the BHATCH command to create an associative hatch object. In this chapter, you learn how to:

- *Specify the pattern to be used and the parameters governing the generation of the pattern.*
- *Define the boundaries of the area(s) to be filled.*

- *Edit a hatch object.*
- *Deal with layers and control the visibility of hatch objects.*
- *Create your own custom hatch pattern.*
- *Use BOUNDARY to create outlines of complex areas.*

Creating Hatch Patterns Using BHATCH

You draw hatch objects whenever you want to highlight an area on your drawing or if you want to convey information pictorially about a specific area on the drawing. For example, you might have a map with different patterns where each pattern represents a distinct type of terrain.

The BHATCH command is used to draw hatch patterns, which are used when you need to fill an area with a repetitive pattern. This section introduces the BHATCH command and also discusses specifying the pattern, defining hatch boundaries, setting attributes, dealing with islands, and using advanced settings.

To issue BHATCH, choose Hatch from the Draw toolbar. The Boundary Hatch dialog box is displayed (see fig. 17.1).

Figure 17.1 *Boundary Hatch dialog box of BHATCH.*

To draw a hatch object, you need to specify the pattern and its parameters, and define the limits of the area to be hatched. These subjects are covered in detail in the following sections.

Specifying the Pattern

The first decision you are faced with is choosing what pattern type to use. You have the option of using one of many predefined patterns, which is the first topic discussed. You can also choose a user-defined pattern, a custom pattern, or imitate an existing hatch object and its parameters. These options are discussed in the sections that follow.

Predefined Patterns

The default pattern type is that of a predefined pattern. AutoCAD comes with a number of predefined patterns. Figure 17.2 shows a sampling of the patterns included with AutoCAD.

Figure 17.2 *Some sample hatch patterns supplied with AutoCAD.*

To select one of the predetermined patterns as your hatch pattern, choose the pattern you want from the Hatch Pattern Palette dialog box, which you display by choosing the Pattern button. When you choose a pattern, a sample of the pattern is displayed in the Pattern Type section of the Boundary Hatch dialog box. You can move on to the next available pattern by picking the sample pattern display. Another option is to select the pattern name from the Pattern drop-down list.

Choosing Scale and Angle Settings

After you choose a pattern, you need to set the Scale and Angle settings. The Scale setting is a scaling factor used to scale the pattern's size up and down, much as LTSCALE is used to control the generation of linetypes. The Angle setting enables you to rotate the pattern.

Some of the patterns are designed to represent real building materials and are defined with the appropriate dimensions. The pattern AR-B88, for example, is used to represent 8" * 8" blocks; using a scale factor of 1, the blocks are drawn as 8" * 8" blocks. Other patterns, such as ANSI31, are simply a symbol or linetype drawn in a repetitive fashion. For patterns representing real building materials, a scale of 1 (or thereabouts) should be used for full-size drawings, whereas the scale used for the symbolic patterns should be related to the plot scale for which the drawing is set up. As a general rule, patterns that represent real building materials have an asterisk in the upper-left corner of the sample display.

[BEEG]TIP: If you set the scale to too small a value, the pattern can take a while to be generated. If you set the scale to too large a value, the pattern can get so big that you cannot even see a portion of it within the area being filled. Use the Preview Hatch button as you fine-tune the parameters.

Metric (ISO) Patterns

Some patterns are designed for use in metric drawings; their names begin with *ISO* and are near the bottom of the list of predefined patterns. As with the metric linetypes, metric hatch patterns should be used only in metric drawings because the patterns are defined in millimeters and would appear too large in drawings set up in English units. If you do choose to use a metric pattern, the ISO Pen Width setting is enabled. Choosing a pen width sets the initial value for the scale setting equal to the chosen pen width, but you are free to set the scale setting yourself.

Linetype and Color Settings

The hatch object is drawn with the current linetype and color settings. If you want to generate the hatch pattern as displayed in the sample box, make sure that the current linetype is continuous or that you are drawing on a layer with a continuous linetype, using a BYLAYER current linetype setting.

User-Defined Patterns

Another type of pattern, a user-defined pattern, is a simple pattern that consists of one or two sets of parallel lines (see fig. 17.3).

The angle and spacing of the first set of parallel lines is set with the Angle and Spacing settings. A second set of parallel lines can be generated perpendicular to the first set by enabling the Double setting, which you can find in the lower-left corner of the Boundary Hatch dialog box. The spacing for the second set of lines is the same as for the first set (see fig. 17.3).

TIP: You can draw a variety of patterns by varying the current linetype with which the pattern is drawn.

Figure 17.3 *Two examples of user-defined patterns.*

Custom Patterns

Additional hatch patterns, similar to the patterns provided with AutoCAD, can also be defined. The additional pattern definitions are either added to the ACAD.PAT file (or ACADISO.PAT), or each definition is stored in its own individual file (named with a .PAT file name extension). The Individual hatch pattern files are referred to as "Custom Pattern" files. To access one of these custom pattern files, choose the Custom Pattern type, and then specify the file name in the Custom Pattern text edit box.

Specifics on how to create your own hatch patterns are discussed in the section "Creating Custom Hatch Patterns," toward the end of this chapter.

Inherit Properties

If you want to duplicate an existing hatch object in the drawing and the settings used to generate it, choose the Inherit Properties button (located in the lower-right quadrant of the Boundary Hatch dialog box) and select the hatch to be duplicated. The settings for the selected pattern are then retrieved and displayed in the Boundary Hatch dialog box.

Defining the Hatch Boundaries

After a pattern and its settings have been selected, the boundaries of the area to be filled must be defined. The area to be filled has to be completely enclosed by one or more objects (see fig. 17.4). You can define the boundaries either by choosing pick points or by selecting the objects. Both methods are discussed in the following sections.

Figure 17.4 *Defining simple areas for hatching with one or more objects.*

In figure 17.4, Area A is defined with a circle (a closed object). Area B is defined with a series of lines that meet end-to-end. Area C is defined with several lines and an arc that cross over each other and do not meet end-to-end. The objects that define the area to be filled are referred to as *boundary objects*.

Using Pick Points

When you choose the Pick Points button in the Boundary Hatch dialog box, the overall hatch area can be delineated automatically by BHATCH. All you have to do is pick a point inside the area that is to be filled; this point is referred to as an *internal point*. With this method, if multiple boundary objects are used to define the hatch area, the boundary objects do not have to meet end-to-end (refer to Area C in fig. 17.4).

Selecting Objects

A less often used option for delineating the area to be hatched is to choose the Select Objects button in the Boundary Hatch dialog box. With this method, you must select the objects that define the area to be hatched. If more than one boundary object exists, the objects must meet end-to-end, as illustrated by Area B in figure 17.4. This method produces erroneous hatch objects with Area C of figure 17.4 because the boundary objects in that area do not meet end-to-end.

Setting the Attributes

By default, the BHATCH command generates an associative hatch object. *Associative* means that the

boundary objects are linked to the hatch object such that when the boundary objects are modified, the hatch object is also modified to conform to the modified boundaries. Furthermore, an associative hatch object's pattern, and the settings used to generate the pattern, can easily be modified with the HATCHEDIT command.

You can choose to disable the Associative setting, but doing so deprives you of the aforementioned advantages when you need to edit the hatch object. Furthermore, you can choose to generate the hatch object as an exploded hatch. By definition, an *exploded hatch* is not an associative object and is not a single object, but rather a collection of lines.

WARNING: Unless you have a very good reason not to, always use the default setting of generating an associative hatch object. Otherwise, the hatch object will not be modified to conform when you modify the boundary objects.

Sometimes the area you want to hatch is an area with subareas within it. These more complicated scenarios are discussed in the next section.

Dealing with Islands

Enclosed areas within the overall area to be hatched are referred to as *islands*. You can even have islands within islands (see fig. 17.5). Text and mtext objects lying within the area to be hatched are also considered islands.

Figure 17.5 *Defining complex areas with islands.*

The pick points method for defining the hatch area automatically detects the islands. If you use the select objects method, you have to explicitly select the boundary objects defining the islands for BHATCH to recognize their presence. The islands themselves are defined with one or more objects, just as you would draw the overall hatch area.

The way islands are treated by BHATCH is controlled by the hatch style setting, which is set by choosing the Advanced button in the BHATCH dialog box and then choosing from the Style drop-down list (see fig. 17.6). The three available styles are Normal, Outer, and Ignore. Figure 17.7 shows how the same islands are treated when using the different styles.

Figure 17.6 *The Advanced Options dialog box of BHATCH.*

Figure 17.7 *Islands using the Normal, Outer, and Ignore hatch styles.*

The default style, Normal, is applicable in most situations. It creates alternating bands of hatching. The Ignore style creates a hatch object that is drawn through the islands. With the Outer style, only the outermost area is hatched. The Outer style is useful when you are hatching overlapping areas with different patterns. Figure 17.8 shows such a situation.

Figure 17.8 *Using the Outer style to hatch overlapping areas.*

One way to approach the problem of hatching overlapping areas is to pick point 1 and, using the Outer style, hatch the outermost area first. Then pick point 2 and the Outer style, and hatch the next area in,

and so on.

BHATCH can also be used on regions. Islands in a region are recognized by BHATCH and are treated according to the current hatch style setting.

In addition to the hatch style settings already discussed, other advanced settings are available to work with in BHATCH.

Choosing Advanced Settings

Choosing the Advanced button displays the Advanced Options dialog box. More often than not, you will not need to change any of these settings.

Define Boundary Set

Normally, the pick points method for delineating the area to be hatched examines all the objects on the screen. You can, however, choose the Define Boundary Set button and explicitly select the objects to be examined for valid hatch boundaries. This option is useful when you have a crowded drawing and want to speed up the algorithm used with the pick points method by restricting the number of objects that are examined.

Style Setting

The Style setting determines how islands are treated. It is discussed earlier, in the "Dealing with Islands" section.

Island Detection Setting

Disabling the Island Detection setting forces the algorithm used with the pick points method to forgo detecting islands. This setting should be left on.

Retain Boundaries Setting

When the hatch area boundary and islands are defined, the areas are automatically delineated with temporary polylines, which normally are removed after the hatch pattern is generated. If you enable the Retain Boundaries setting, the temporary polylines are not removed and are drawn on the current layer. You can even choose to retain the polylines as a region. Enabling the Retain Boundaries option is most useful when the area is delineated by multiple objects and you want a single polyline or region to represent the hatch area. If you subsequently use the AREA command on the resulting polyline(s), or MASSPROP on the resulting region, you can easily determine the area.

The following exercise takes you through the necessary steps of using BHATCH to fill some areas on several details with a pattern.

FILLING IN AREAS WITH BHATCH

1. Open the drawing SOLID.DWG. This drawing contains a 3D solid model. The view you initially see is of paper space with several viewports defined. The viewports themselves are on layer VPORTS, which is frozen.

2. Restore the view ENDVIEW. Make sure that you are in floating model space (double-click on the Paper button, if necessary). Make the Right End view the current viewport by clicking anywhere in that detail. Make a new layer, HATCH, and make sure that it is the current layer.

3. Choose Hatch from the Draw pull-down menu, choose the pattern ANSI34, and set the scale to 1 and the rotation angle to 0. Choose the Pick Points button and pick 1 (see fig. 17.9). Click on the Apply button. The results should look like figure 17.9.

Figure 17.9 *Hatching the details.*

4. Make sure you are in paper space and restore the view SECTION, make floating model space current, and repeat the BHATCH command using the same pattern and parameters as before. Click on the Pick Points button and pick 2 and 3 (refer to fig. 17.9). Click on the Apply button. The results should look like figure 17.10.

5. Save the drawing; you are finished with this exercise.

Figure 17.10 *SOLID.DWG with the hatch objects drawn.*

Editing Hatch Objects

To edit hatch objects, issue the HATCHEDIT command by choosing Hatch from the Object submenu of the Modify pull-down menu. The resulting Hatchedit dialog box is similar to the Boundary Hatch dialog box, but with several settings disabled (see fig. 17.11).

Figure 17.11 *The Hatchedit dialog box.*

With HATCHEDIT, you can change the pattern of the hatch object or the parameters that control the generation of the pattern. You also can access the Hatchedit dialog box by choosing the Hatch Edit button in the Modify Hatch dialog box (DDMODIFY command).

Editing the Hatch Boundaries

If you stretch or move the boundary objects defining the overall area of an associative hatch object, the hatch object automatically adjusts to fit the modified boundaries. If you move, delete, or stretch any of the islands, the hatch object also is adjusted.

If you delete any of the boundary objects defining the overall hatch area or islands (resulting in an open rather than closed area), the associativity is removed from the hatch object and the hatch loses its capability to adjust to changing boundaries.

WARNING: Islands should never be moved beyond the outermost hatch boundary. If you ignore this advice and do so anyway, you may encounter problems later, such as the hatch adjusting incorrectly, move as you continue to edit the hatch or the boundary objects.

Exploding Hatch Objects

You can explode a hatch object into its constituent lines with the EXPLODE command. Exploding a hatch object removes the associativity of the object. Additionally, the single hatch object is replaced by the group of line objects that make up the pattern. Exploding a hatch object does enable you to edit the individual lines of the hatch, but in most cases you lose more than you gain.

Using Object Snap

A hatch object is composed of lines. Therefore, you can use the same object snap modes (such as endpoint or midpoint) on the lines in an associative hatch object as you use on lines.

NOTE: In earlier versions of AutoCAD, a hatch was actually a form of a block, and you could use the insert object snap mode on a hatch. In R14, a hatch is an object in itself and has no insertion point.

The following exercise shows how to modify the hatch objects that you drew in the previous exercise.

STRETCHING THE BOUNDARIES OF THE HATCH OBJECT

1. Continue to use the drawing SOLID.DWG found in the Chapter 17 folder on the accompanying CD-ROM. Restore the view ENDVIEW. Make sure that you are in floating model space and that the current viewport is that of the End View.

2. Choose Hatch from the Objects submenu of the Modify pull-down menu, select the hatch object, change the pattern to ANSI35, and click on the Apply button.

3. Restore the view SECTION. Make sure that the current viewport is that of the Section view. Choose Stretch from the Modify toolbar, and choose 1 and 2 (see fig. 17.12).

4. Type in the displacement **-0.5,0** for the first stretch point, and press the Enter key for the second stretch point.

5. You are finished with this drawing; save and exit the drawing.

Figure 17.12 *Stretching the boundaries.*

Dealing with Layers and Visibility

There are two methods with which you can control the visibility of hatch objects, layers or the system variable FILLMODE. Both are discussed in the following sections.

Controlling Visibility with Layers

Quite often, the hatch object is drawn on a layer separate from the layer(s) containing the boundary objects. Drawing the hatch object on a separate layer enables you to make the hatch object invisible, while leaving the boundary objects visable, by freezing or turning off the hatch layer. When the time comes to update the hatch object, however, certain consequences must be dealt with because the hatch boundaries have been modified. When you lock the hatch layer and change the boundary objects some

consequences will result. The following sections discuss these consequences.

If the hatch layer is frozen and the boundary objects are modified, then the associativity of the hatch object is removed and the hatch object cannot adjust to the changed boundaries.

If the hatch layer is turned off and the boundary objects are modified, the hatch object still adjusts to the changed boundaries. The adjustment is evident when the hatch layer is turned back on.

If the hatch layer is locked and the boundary objects are modified, the hatch object is not adjusted to the changed boundaries; however, the hatch object's associativity remains intact.

To force the hatch object to adjust to the modified boundary objects, first unlock the hatch layer. Then select the hatch object with the HATCHEDIT command and click on the Apply button without making any changes to the settings. If at any time the hatch object fails to adjust correctly to the modified boundaries, you can force AutoCAD to try again by selecting the hatch object with the HATCHEDIT command and clicking on the Apply button without changing any of the parameters.

TIP: To avoid accidentally removing the associativity from hatch objects, get into the habit of turning off the hatch layer rather than freezing it.

Controlling Visibility with FILLMODE

You can control the visibility of all hatch objects in a drawing by setting the FILLMODE system variable. When FILLMODE is turned off (set to 0), all hatch objects become invisible, regardless of the status of the layers on which the hatch objects reside. Of course, you must issue the REGEN or REGENALL command to affect existing hatch objects. The disadvantage of using FILLMODE is that FILLMODE also affects solids, multilines, and wide polylines.

Completely Filling an Area

NOTE: In earlier versions of AutoCAD, when you wanted to completely fill an area (coloring an area, for example), you either drew solids or generated a hatch with settings that created a dense pattern of hatch lines. In R14, a new pattern has been provided to take care of filling areas: the solidfill pattern. Using the solidfill pattern is much more efficient, in terms of plotting and regenerating the drawing, than the old method of drawing a hatch with settings that result in a dense pattern.

Selecting Hatch Objects

In selecting an associative hatch object, you can choose not only to select the hatch object itself but also to have the boundary objects associated with the hatch object automatically included in the selection. In the Object Selection Settings dialog box (displayed by choosing Selection from the Tools pull-down menu), is a setting labeled Associative Hatch. By default, this option is disabled so that when you select a hatch object, the associated boundary objects are not selected automatically. If you enable the Associative Hatch option, the boundary objects associated with the selected hatch object are

automatically included in the selection.

Selecting a hatch object's boundary objects without selecting the hatch object itself can be difficult to do without magnifying the view, turning off the hatch object's layer, or turning FILLMODE off and regenerating the drawing. Another tool is available, however, that you can use to select the boundary objects that can be more efficient at time--*object cycling*.

Using the Direct Hatch Option of HATCH

The older version of the BHATCH command is HATCH. Because it is the older version, it is not located on any of the pull-down menus or toolbars, and must be typed. The major drawback to using HATCH is that it can draw only nonassociative hatch objects.

Despite this, HATCH does have an option that you may find useful: the Direct Hatch option. The Direct Hatch option enables you to define on-the-fly the area to be hatched, removing the necessity to draw the boundary objects before drawing a hatch object. The direct option is most useful when you have a large area to hatch and want to hatch representative patches, not the entire area.

After you issue the HATCH command, choose a pattern, and set the associated settings, you are prompted to select the boundary objects. To invoke the Direct Hatch option, do not select any objects. Press Enter. Then you define the hatch boundary by using options similar to those in the PLINE command. In effect, you are drawing the hatch boundary with a temporary polyline. After you finish defining the area with a closed polyline, the nonassociative hatch object is drawn. If you want, you can even choose to retain the polyline.

In the next section, you learn how to create your own hatch patterns.

Creating Custom Hatch Patterns

It is possible to add new patterns to the ones supplied with AutoCAD. You can add the new patterns (custom hatch patterns) to the file ACAD.PAT (or ACADISO.PAT) or define each new pattern in its own file. ACAD.PAT and ACADISO.PAT are both found in the \ACADR14\SUPPORT directory (also referred to as custom pattern files). If you choose to define each pattern in its own file, the file must have the same name as the pattern and have a file name extension of .PAT. The new custom patterns files should be placed in one of the directories defined in the support file search path (see the PREFERENCES command). Because hatch pattern files are ASCII files, a text editor is all you need to add to ACAD.PAT or to create your own file.

A hatch pattern consists of one or more families of parallel pattern lines. The rules for defining a pattern line are the same as those for defining a new linetype, except that no text or shapes can be included in the definition of the pattern line.

TIP: Although the rules for defining a hatch pattern are relatively straightforward, implementing the rules takes time, effort, and patience. A much easier and more cost-efficient solution is to buy the pattern you need from one of several third-party developers (check with your AutoCAD dealer). If you definitely want to define your own pattern, read on.

Defining the Header Line

The first line in any pattern definition is the header line:

```
*pattern-name [, description]
```

The name cannot contain any blanks. The description is optional (as is the preceding comma), and is only used by the ? option of the HATCH command.

NOTE: If you choose to place the pattern name in its own custom pattern file, you must use the same name for both the file and the pattern.

Defining the Pattern Lines

The header line is followed by one or more pattern line descriptors, one for each family of lines to be drawn, with the following syntax:

```
angle, x-origin, y-origin, delta-x, delta-y [,dash-1, dash-2, ...]
```

The following line descriptor, for example, would result in the hatch shown on the left in figure 17.13:

```
*L45, 45 degree lines @ 0.25 units apart

45,0,0,0,0.25
```

Figure 17.13 *Samples of the L45 and TRIANG patterns.*

Each family of lines starts with one line, and the line's angle and origin are specified by the first three numbers of the line descriptor. In the preceding example, the first line is drawn at a 45-degree angle through the point 0,0. The family of lines is generated by offsetting each successive line by delta-x and delta-y offsets, with delta-x measured along the line and delta-y measured perpendicular to the lines. In the example, each succeeding line is offset 0 in the x direction and 0.25 in the y direction. With no other dash specifications specified, AutoCAD draws the lines with the current linetype.

The next example is a pattern taken from the file ACAD.PAT, and is shown on the right in figure 17.13:

```
*TRIANG, Equilateral triangles

60, 0,0, .1875,.324759526, .1875,-.1875
120, 0,0, .1875,.324759526, .1875,-.1875
0, -.09375,.162379763, .1875,.324759526, .1875,-.1875
```

In this example, the pattern consists of three families of lines: one family at 60 degrees, another at 120 degrees, and the third at 0 degrees. The dash specifications (the last two numbers in each line) indicate that each line is to consist of a 0.1875 dash and a 0.1875 space repetitive pattern.

You can have as many pattern line descriptors as you want, but each line can be no more than 80 characters long.

Adding Sample Slides

The samples of the patterns displayed in the Boundary Hatch dialog box are slides that are stored in the slide library file ACAD.SLB. If you want a sample of a custom pattern you have added to the file ACAD.PAT to be displayed in the dialog box, you must draw a sample of the pattern, make a slide of the drawing, and add the slide to the ACAD.SLB file (also found in the \ACADR14\SUPPORT directory). The slide file must have the same name as the pattern it represents. See Chapter 23, "Creating Scripts and Slide Libraries," for the commands and procedures used to create slides.

You can use the program SLIDELIB, which is supplied with AutoCAD, to recreate the slide library file ACAD.SLB with your additional new slides. Unfortunately, SLIDELIB cannot be used to add new slides to a library file. In effect, you must recreate the entire library file when you add to it. To recreate ACAD.SLB, you need the original slide files that were used to create ACAD.SLB (and these slide files are not supplied with AutoCAD).

Several third-party packages are available, however, that make managing, deleting from, and adding to a slide library file easier to do.

NOTE: One such package, a DOS program called MAKSLB21, is provided on the accompanying CD in the Chapter 17 folder. Contact your AutoCAD dealer to see what other packages are available.

Using BOUNDARY to Delineate Areas and Islands

The BOUNDARY command is a variation of the BHATCH command.

BOUNDARY is used to create polylines delineating an overall area and the islands within that area. If you want, you can create a region rather than polylines. The boundary delineation algorithm employed by BOUNDARY is the same as the algorithm employed by BHATCH. The Boundary Creation dialog box is a variation of the Advanced Options dialog box of the BHATCH command (see fig. 17.14). Use Boundary when all you want to do is to delineate an area and its islands but not hatch the area.

Figure 17.14 *Boundary Creation dialog box of BOUNDARY.*

As with the BHATCH command, the overall area and islands must be enclosed by one or more objects.

Summary

Hatching is a powerful tool for clarifying the meaning of your drawing or for conveying information to the reader. It is easy to apply, using BHATCH, and just as easy to edit with HATCHEDIT. Drawing hatch objects on a separate layer is always a good idea. Although you can create your own hatch patterns, buying the pattern you need is usually more cost-effective. In the next chapter, "Productive Dimensioning," you learn about another group of associative objects, the group of dimension objects.

← Previous Chapter → Next Chapter ↑ Contents